INF 533 - Programação para a Web
Abril - Junho 2009

Prof. Célio Guimarães ***IC - Unicamp***

Atualizado em 27 - Abril -2009

Ementa: 1ª Parte: programação no Cliente

Introdução às linguagens HTML e XHTML via exemplos.
Introdução a XML: histórico, conceitos básicos, DTDs. Diferenças entre HTML e XHTML. Validação de documentos HTML 4.0 e XHTML através de utilitários disponibilizados na w3.org.
A influência de XML sobre HTML: Cascading Style Sheets (CSS) - conceitos e exemplos.
Javascript através de exemplos: programação, eventos, formulários; discrepâncias no tratamento de eventos básicos entre Firefox e Internet Explorer.
Expressões regulares; validação de dados de formulários em Javascript via expressões regulares.
Introdução à técnica de "layout via tableless" preconizada pela w3.org.
Introdução aos protocolos http 1.0 e 1.1. O problema de "estado".

Referências para leitura e estudo:

Livro do Deitel que cobre quase todo o material da disciplina: Internet & World Wide Web How to Program (a livraria Cultura tem o livro traduzido para Português; não sei dizer a qualidade da tradução).
"Biblia" de HTML: "HTML & XHTML: The Definitive Guide", 5ª Ed., Chuck Musciano and Bill Kennedy (O'REilly, Paperback - Aug 2002)
Introdução a linguagens de marcação (incluído na apostila). Exemplos em HTML e XML do livro "XML How to Program", Deitel et al, Pr. Hall 2001 (Capítulos 1 - 6: serão abordados no curso e deverão ser trabalhados/modificados pelos alunos).
Editor HTML para Windows (livre): First Page. (inclui o sistema ***Tidy*** de Dave Raggett para conversão de um documento HTML obedecendo à especificação HTML 4.0 ou XHTML).
Tutoriais incluídos na apostila sobre: HTML de Dave Raggett; s Cascading Style Sheets (CSS) do mesmo autor e um tutorial abrangente sobre CSS.
Símbolos especiais em HTML.
Nomes de cores reconhecidos por Mozilla/Firefox e I. Explorer (mas não pela w3.org).
Tutorial interativo sobre Javascript (cópia na apostila).
Tudo que V. queria saber sobre Javascript: guia Mozilla Javascript
Eventos suportados em Javascript 1.3 (veja um resumo na apostila)
Excelente tutorial sobre expressões regulares (veja também na apostila).
Exemplos de validação de campos via expressões regulares em Javascript.
Técnica *tableless* para projeto de páginas formatadas com colunas sem usar frames ou tabelas, preconizada pela w3.org e baseada em folhas de estilo (a página tem um link para o mesmo documento traduzido para português). O mesmo exemplo acima, ligeiramente simplificado (veja no diretório correspondente as folhas de estilo usadas na implementação da página). Outro exemplo de página formatada em 3 colunas, sem tabelas
Usabilidade: Como usuários lêm na WEB (do guru Jakob Nielsen). Alguns conselhos para planejamento visual de páginas de dois especialistas: Robin Williams e Jakob Nielsen.

Projeto da 1ª parte: desenvolvimento de uma aplicação simples envolvendo formulários, tabelas, CSS, Javascript, expressões regulares e layout "tableless", em conformidade com os padrões HTML 4.01 ou XHTML (esta aplicação servirá de *front-end* para o projeto da 2ª Parte).
Atenção: você deverá usar um (ou mais) arquivo(s) de folhas de estilo (CSS) para todas as páginas do projeto e não deverá colocar estilos in-line nos seus documentos.
Deverá também usar a técnica de *layout tableless* na página principal do seu projeto (usando duas ou três

colunas de sua livre escolha). Não deixe de estudar pelo menos um artigo sobre usabilidade como este do Jakob Nielsen.
Inclua na página principal os nomes dos participantes e um link para uma ou duas páginas contendo: **(i)** uma pequena descrição do seu projeto, manual de utilização **(ii)** detalhes da implementação incluindo características relevantes, técnicas utilizadas, etc. Caso seu projeto não esteja num servidor da Unicamp, comunique ao professor e envie via e-mail todos os arquivos utilizados.
Prazo de entrega:

Exercícios

1. A fim de melhorar a apresentação visual, coloque numa tabela com duas colunas o formulário contido no exemplo do Deitel). A 1ª coluna deve conter a descrição de cada campo e a 2ª coluna o campo para entrada de dados.

2. Modifique o exercicio anterior colocando comandos de estilo CSS no cabeçalho da página e retirando-os do corpo do documento. Valide a sua página no site de validação da w3.org

3. Exercício sobre *layout tableless* :
   Modifique o exemplo básico apresentado em *layout tableless* de forma a gerar uma "homepage" com *2 colunas*, onde a coluna da esquerda mostra o logo da unicamp (é o mesmo no cabeçalho desta página - atenção para as dimensões) seguido do conteúdo (body) da página nav.html e a coluna da direita apresenta o logo do IC alinhado à direita, seguido do conteúdo da página main.html (onde o texto de main.html deve fluir em torno do logo do IC), esta por sua vez separada por um linha horizontal do conteúdo da página list.html (para detalhes sobre o projeto de caixas *(div element)* veja o capítulo 7 do tutorial sobre CSS da Oxfod Brookes University). Uma impressão da página resultante pode ser vista na apostila da disciplina imediatamente após o documento "Tableless layout HOWTO".
   Observe que a técnica deste exercício permite, de forma muito simples, simular uma janela com três frames!
   Obs: (i) acrescente o(s) nome(s) do(s) autor(es) no final ou no início da página.
   (ii) Teste a sua página com o Internet Explorer e Mozilla (Firefox) e ajuste eventuais diferenças!
   (**sugestão:** faça os seus testes iniciais colocando bordas nas 2 caixas)
   Submeta uma impressão do arquivo html/css e da página mostrada pelo navegador.
   Prazo de entrega:

4. Exercícios sobre validação de campos de formulários usando expressões regulares em Javascript.
   Prazo de entrega:
5. **Processamento de dados de formulários em php:**
   Escreva um programa em php que pega os dados do formulário do exercício anterior (nome, data de nascimento e telefone) e gera uma página html contendo uma tabela com 2 colunas e 3 linhas, onde cada linha contém o nome do campo e o valor do mesmo. O formulário deverá ser submetido usando o método POST. Exemplo:

   ```
   form <name="myform" method="POST" action= "cgi-bin/ex5.php">
   . . .
   neste exemplo o formulário está no subdiretório public_html e o
   o programa ex5.php no subdiretório public_html/cgi-bin
   ```

   Para obter o valor de um campo submetido pelo formulario, digamos, *telefone*, V. deve escrever um comando em php do tipo:

   ```
   $fone = $_POST['telefone'];
   ```

   onde telefone é o nome dado ao elemento correspondente no formulário.
   Prazo de entrega:

6. **Armazenamento e recuperação de dados de formulários em arquivo *dba*:**
   Obtenha os dados do formulário do problema anterior e armazene-os num registro de um arquivo dba, usando o caracter "|" como separador. Em seguida feche o arquivo e abra-o para leitura; leia o registro e

através da função **explode** coloque os campos num array php. Em seguida use o código do exercício anterior para gerar uma tabela html e exibi-la no browser.
Opcional: em vez de abrir o arquivo para leitura no passo 2 acima, abra-o para leitura/escrita, leia o registro, atualize apenas um campo do registro (gravando de volta no arquivo dba) e gere a tabela html contendo o registro atualizado.
Prazo de entrega:

Ementa: 2ª Parte: programação no Servidor

Conceitos básicos de um Servidor Web: o Servidor Apache. Introdução à linguagem *script* Php de programação para a Web: semelhanças com a linguagem C; estruturas de dados, comandos, integração com o protocolo CGI. Exemplos interativos do *bluebook*.
O conceito de "sessão"; problemas com a característica de "falta de estado" *(stateless)* do protocolo http; soluções com *cookies* e outras técnicas para introduzir estado na aplicação.
Interface de Php com um gerenciador de base de dados relacional (MySQL); exemplos;
Introdução às técnicas **AJAX** (*Asynchronous Javascript and XML*). Exemplos.

Referências para leitura e estudo (programação no Servidor):

Tutoriais (muito bons) do James Marshall. Em particular veja o tutorial sobre HTTP e um resumo do mesmo.
Outro tutorial sobre HTTP. Introdução aos *tipos de conteúdo* MIME.
Testando / modificando os exemplos em PHP do "livro azul":
Na URL: **scripts do bluebook** Você encontrará diretórios com os scripts do "livro azul"; clique no link do script php que V. quer testar (o nome diz a página do livro em que ele se encontra). Os scripts foram alterados para mostrar o código fonte em php seguido da vizualização do resultado da sua execução.
Obs: o arquivo inicial do diretório, **chn.txt**, contém o código fonte de todos os scripts do capítulo **n**. Basta copiar e colar na sua área para testá-los, modificá-los, etc. diretamente a partir da sua área.
Capítulo do livro "Web Database Applications", Hugh Williams, sobre controle de sessões em PHP.
Um exemplo muito simples de sessão com uma única variável de estado: contador de visitas . Veja a seguir o código fonte.
Mais um exemplo de sessão em Php envolvendo uma mini-aplicação com 3 páginas: pform.html, thanks.php e regform.php. Obs: a aplicação foi elaborada para funcionar mesmo quando o "browser" do cliente desabilita cookies, conforme detalhado em aula.
Um exemplo mais elaborado de contrôle de sessão: o conhecido jogo da forca, adaptado para nomes femininos em Português. Utiliza 5 variáveis de estado.

Projeto da 2ª Parte: desenvolvimento de uma aplicação no Servidor envolvendo estado, com dados persistentes armazenados num SGBD relacional ou em arquivos indexados db3, utilizando a interface da 1ª Parte.
O seu sistema deve suportar **inserção, atualização, consulta e remoção** de dados da BD. O seu projeto deve ser capaz de ser inteiramente testado via Web. Envie um arquivo *.zip* contendo todos os fontes.
Sugestão: Para cifrar senhas use a função *md5()* ou *crypt()*.
Prazo de entrega: a ser definido

Como usar PHP no ambiente do IC:

1. dê login via ssh na servidora do IC3 (ssh -l seulogin ssh.students.ic.unicamp.br);
2. se preciso, coloque o seu homedir com direito de acesso x para todos (chmod a+x ~)
3. crie o diretório **public_html** com direitos de acesso rx para todos (chmod a+rx public_html)
4. crie um subdiretóro de public_html, digamos, **cgi-bin** também com direitos de acesso rx para todos.
5. coloque em cgi-bin seus programas php, começando com, digamos, **teste.php** (com direitos de leitura para todos)
6. no seu browser preferido digite a URL: **http://bidu.lab.ic.unicamp.br/~nome_login/cgi-bin/teste.php**
   V. deverá ver na tela do browser a saída emitida pelo seu programa teste.php

7. para ter acesso ao valor de um campo de formulário cujo nome é, digamos, *email* Você deve usar o array global $_POST como se segue: *$email= $_POST['email']*
8. um exemplo simples de um script PHP que critica os dados de um formulário em 3 fases (execuções) distintas:
   - 1ª execução: envia o formulário para o browser;
   - 2ª execução: recebe e critica os dados do formulário, e, se incorretos, envia uma mensagem de erro para o browser e re-envia o mesmo formulário para resubmissão;
   - 2ª ou 3ª execução: se dados estão corretos processa-os de forma apropriada e envia os resultados para o browser numa página formatada em HTML.

Bases de dados para o projeto da 2ª Parte:

Implemente sua Base de Dados usando uma das duas opções a seguir:

1. **Criação/atualização de arquivos indexados usando a biblioteca dba do PHP (preferivel)**

   Um exemplo simples. Veja a sua execução na servidora bidu.lab. O seu arquivo indexado deverá estar num diretório com permissão de escrita para todos (pois será criado/atualizado pelo módulo PHP sob o usuário *nobody* do Servidor Web). Para este fim (e por razões de segurança) crie um subdiretório, digamos, *dados* do seu diretório *cgi-bin* e faça *chmod a+rw dados* e crie o seu arquivo indexado nesse diretório (veja como foi feito no exemplo)
   Obs importante:um erro comum observado nos laboratórios é utilizar o módulo PHP instalado no servidor Web do IC, ou seja, como a seguir: **http://www.students.ic.unicamp.br/~nome_login/cgi-bin/teste.php**: o seu programa *teste.php* não terá acesso à biblioteca dba de arquivos indexados nem aos SGBDs relacionais do IC (ambos só disponíveis na **bidu.lab**!).
   Documentação da biblioteca dba (link para o manual on-line em *php.net*).

2. **Utilização de SGBD Relacional disponível na bidu.lab via biblioteca Adodb ou MySql escolha esta opção somente se V. conhece SQL!**

   Para criar e visualizar tabelas num dos SGBDs Relacionais da *bidu.lab* execute este utilitário PHP. Um pacote mais elaborado para o mesmp fim pode ser usado na bidu.lab: phpMyAdmin Ele ilustra o uso da biblioteca *Adodb* para acesso via comandos SQL aos Sistemas Gerenciadores de Base de Dados (SGBD) Relacionais disponíveis na servidora *bidu.lab* (use o *Postgres* ou *MySql*). Veja aqui o código fonte simplificado: sem o formulário para entrada de dados e com uma consulta default.